IN THE CLAIMS

Please amend the claims as follows:

1. (currently amended) A data structure stored on a computer-readable medium ~~having stored thereon a data structure~~ for use in a computer-aided design and verification system for interconnecting instrumentation logic in a simulation model of a compiled digital circuit design that includes one or more design entities described utilizing a hardware description language (HDL), said data structure comprising:

~~an~~ a first instrumentation ~~declaration comment~~ ~~containing data representing a cross-hierarchical instrumentation~~ entity descriptor field containing data representing a first instrumentation entity, the data representing said first instrumentation entity including a non-conventional HDL comment port mapping syntax processed by a post-compiler instrumentation load tool to instantiate said first instrumentation entity within at least one of said one or more design entities, wherein said non-conventional HDL comment port mapping syntax is recognized by an HDL compiler such that the HDL compiler does not instantiate said first instrumentation entity into the digital circuit design, said data representing said first instrumentation entity using the non-conventional comment port mapping syntax further including a simulation event descriptor field containing data representing a simulation event having a designated event name and that is generated by said first instrumentation entity responsive to one or more input signals from said at least one of said design entities; and

~~an input port mapping comment containing data representing a simulation event that is input into said cross-hierarchical instrumentation entity to generate a cross-hierarchical simulation event~~ a second instrumentation entity descriptor field containing data representing a second instrumentation entity instantiated within at least one of said one or more design entities utilizing said non-conventional comment port mapping syntax, said second instrumentation entity descriptor field comprising an input port mapping field containing data for referencing the generated simulation event to an input port of said second instrumentation entity utilizing an extended event identifier, said extended event identifier including an event type identifier field that specifies a simulation event type and an event name field that includes said designated event name.

2. (currently amended) The ~~computer-readable medium~~ data structure of claim 1, wherein said input port mapping ~~comment~~ field further comprises[[:]]

an instance identifier field containing data representing a ~~hierarchical~~ list of design entities arranged in descending hierarchical order in which said generated simulation event occurs~~; and~~

~~an event identifier field containing data representing an instrumentation entity and that generates said simulation event~~.

3. (currently amended) The ~~computer-readable medium~~ data structure of claim 2, wherein said instance identifier field further comprises:

data representing a highest level design entity in which said ~~cross-hierarchical~~ first instrumentation entity is instantiated;

data representing a lowest level design entity in which said generated simulation event occurs; and

data representing intermediate design entities between said highest level design entity and said lowest level design entity.

4. (Cancelled)

5. (currently amended) The ~~computer-readable medium~~ data structure of claim [[2]] 1, wherein said extended event identifier ~~field~~ further comprises[[:]]

~~a first event identifier sub-field~~ an instrumentation entity instantiation name field containing data ~~representing~~ specifying an instance name of said first instrumentation entity~~; and~~

~~a second event identifier sub-field containing data representing an event type; and~~

~~a third event identifier sub-field containing data representing an instance of said event~~.

6. (currently amended) ~~A~~ In a computer-aided design and verification system, a method for ~~instrumenting a cross-hierarchical simulation event, wherein said cross-hierarchical simulation event is a function of a first simulation event residing at a first level of simulation model hierarchy and a second simulation event residing at a second level of simulation model hierarchy, wherein said first level of simulation model hierarchy is not at a lower level of said~~

~~simulation model hierarchy than said second level of simulation model hierarchy~~ interconnecting instrumentation logic in a simulation model of a compiled digital circuit design that includes one or more design entities described utilizing a hardware description language (HDL), said method comprising:

~~defining a cross-hierarchical instrumentation entity within said first level of simulation model hierarchy~~ describing a first instrumentation entity using a non-conventional HDL comment port mapping syntax processed by a post-compiler instrumentation load tool to instantiate said first instrumentation entity within at least one of said one or more design entities, wherein said non-conventional HDL comment port mapping syntax is recognized by an HDL compiler such that the HDL compiler does not instantiate said first instrumentation entity into the digital circuit design, said description of said first instrumentation entity using the non-conventional comment port mapping syntax further including a simulation event descriptor field containing data representing a simulation event having a designated event name and that is generated by said first instrumentation entity responsive to one or more input signals from said at least one of said design entities; and

~~connecting a first input of said instrumentation entity to said first simulation event and connecting a second input of said instrumentation entity to said second simulation event~~ describing a second instrumentation entity utilizing said non-conventional comment port mapping syntax, said description of said second instrumentation entity including an input port mapping field containing data for referencing the generated simulation event to an input port of said second instrumentation entity utilizing an extended event identifier, said extended event identifier including an event type identifier field that specifies a simulation event type and an event name field that includes said designated event name.

7-9. (Cancelled)

10. (newly added) The method of claim 6, wherein said input port mapping field further comprises an instance identifier field containing data representing a list of design entities arranged in descending hierarchical order in which said generated simulation event occurs.

11. (newly added) The method of claim 10, wherein said instance identifier field further comprises:

data representing a highest level design entity in which said first instrumentation entity is instantiated;

data representing a lowest level design entity in which said generated simulation event occurs; and

data representing intermediate design entities between said highest level design entity and said lowest level design entity.

12. (newly added) The method of claim 6, wherein said extended event identifier further comprises an instrumentation entity instantiation name field containing data specifying an instance name of said first instrumentation entity.

13. (newly added) A computer program product stored on a computer-readable medium for use in a computer-aided design and verification system for interconnecting instrumentation logic in a simulation model of a compiled digital circuit design that includes one or more design entities described utilizing a hardware description language (HDL), said computer program product comprising:

a first instrumentation entity descriptor field containing data representing a first instrumentation entity, the data representing said first instrumentation entity including a non-conventional HDL comment port mapping syntax processed by a post-compiler instrumentation load tool to instantiate said first instrumentation entity within at least one of said one or more design entities, wherein said non-conventional HDL comment port mapping syntax is recognized by an HDL compiler such that the HDL compiler does not instantiate said first instrumentation entity into the digital circuit design, said data representing said first instrumentation entity using the non-conventional comment port mapping syntax further including a simulation event descriptor field containing data representing a simulation event having a designated event name and that is generated by said first instrumentation entity responsive to one or more input signals from said at least one of said design entities; and

a second instrumentation entity descriptor field containing data representing a second instrumentation entity instantiated within at least one of said one or more design entities utilizing

said non-conventional comment port mapping syntax, said second instrumentation entity descriptor field comprising an input port mapping field containing data for referencing the generated simulation event to an input port of said second instrumentation entity utilizing an extended event identifier, said extended event identifier including an event type identifier field that specifies a simulation event type and an event name field that includes said designated event name.

14. (newly added) The program product of claim 13, wherein said input port mapping field further comprises an instance identifier field containing data representing a list of design entities arranged in descending hierarchical order in which said generated simulation event occurs.

15. (newly added) The program product of claim 14, wherein said instance identifier field further comprises:

data representing a highest level design entity in which said first instrumentation entity is instantiated;

data representing a lowest level design entity in which said generated simulation event occurs; and

data representing intermediate design entities between said highest level design entity and said lowest level design entity.

16. (newly added) The program product of claim 13, wherein said extended event identifier further comprises an instrumentation entity instantiation name field containing data specifying an instance name of said first instrumentation entity.